AFRL-SN-WP-TR-2001-1101

CONTEXT AND QUASI-INVARIANTS IN AUTOMATIC TARGET RECOGNITION (ATR) WITH SYNTHETIC APERTURE RADAR (SAR) IMAGERY

Thomas O. Binford
Tsung-Liang Chen

Stanford University
651 Serra Street
Stanford, CA 94305

AUGUST 2000

FINAL REPORT FOR PERIOD 03 JUNE 1997 – 03 JUNE 2000
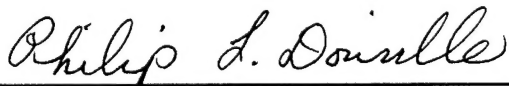
20020329 066

NOTICE

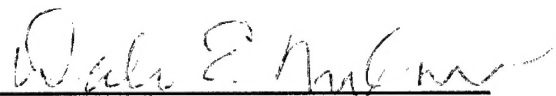USING GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA INCLUDED IN THIS DOCUMENT FOR ANY PURPOSE OTHER THAN GOVERNMENT PROCUREMENT DOES NOT IN ANY WAY OBLIGATE THE US GOVERNMENT. THE FACT THAT THE GOVERNMENT FORMULATED OR SUPPLIED THE DRAWINGS, SPECIFICATIONS, OR OTHER DATA DOES NOT LICENSE THE HOLDER OR ANY OTHER PERSON OR CORPORATION; OR CONVEY ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY RELATE TO THEM.

THIS REPORT IS RELEASABLE TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

Philip L. Douville, Ph. D.
Project Engineer
Target Recognition Branch

Dale E. Nelson, Ph. D.
Chief, Target Recognition Branch
Sensor ATR Technology Division
Sensors Directorate

DAVID W. CHANDLER, Lt Col, USAF
Deputy, Sensor ATR Technology Division
Sensors Directorate

Do not return copies of this report unless contractual obligations or notice on a specific document requires its return.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, Paperwork Reduction Project (0704-0188) Washington DC 20503
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) August 2000	2. REPORT DATE FINAL	3. DATES COVERED: *(From – To)* 06/03/97-06/03/00

4. TITLE AND SUBTITLE

Context and Quasi-Invariants in Automatic Target Recognition (ATR) with Synthetic Aperture Radar (SAR) Imagery

5a. CONTRACT NUMBER
F33615-97-1-1016

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62204F

6. AUTHOR(S)

Thomas O. Binford
Tsung-Liang Chen

5d. PROJECT NUMBER
ARPA

5e. TASK NUMBER
AA

5f. WORK UNIT NUMBER
1R

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Stanford University
651 Serra Street
Stanford, CA 94305

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Sensors Directorate
Air Force Research Laboratory
Air Force Materiel Command
Wright-Patterson AFB, OH 45433-7318

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/SNAT

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-SN-WP-TR-2001-1101

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release, distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This report focuses on the development of an automatic target recognition (ATR) system using high resolution synthetic aperture radar (SAR) imagery. The system achieves 95 to 100 percent recognition rates when applied to a set of MSTAR images. Typically, the system takes less than one minute to match an input image to a candidate vehicle class with Matlab programs running on a Pentium II 300 MHz machine. Experiments based on conventional recognition techniques were conducted for comparisons. Study of persistent scattering confirms the feasibility of implementing a SAR ATR system using physical image features. A new generic vehicle model, parameterized by the length, width, and orientation of a target is used in a two-phase recognition process with hypothesis generation and verification aimed at addressing the combinatorial target recognition problem. In the hypothesis generation stage, a few likely candidate target classes are identified from a target database with positive evidence. The candidates are assessed using both positive and negative evidence in the hypothesis verification stage. Leading surface estimation, image alignment, Delaunay walk, and recognition metrics are introduced to improve performance of the SAR ATR system.

15. SUBJECT TERMS
Automatic target recognition (SAR ATR), image recognition, image alignment, and computational geometry.

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT: SAR	18. NUMBER OF PAGES 60	19a. NAME OF RESPONSIBLE PERSON (Monitor) Phil Douville
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			19b. TELEPHONE NUMBER *(Include Area Code)* (937) 255-1115 x4378

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18

HES&S 31-15093-1

i

Contents

List of Figures

List of Tables

1. Early Recognition Experiments

At the beginning phase of the project, we carried out a series of experiments on recognition using conventional techniques including K-L decomposition (eigenimage approach) and simple template matching. We also tested configuration-independent techniques with target dimensions. These results were intended for the understanding of synthetic aperture radar automatic target recognition (SAR ATR) and for comparisons.

A set of 200 chips for each of the 3 targets (BTR70, BMP2, and T72) are used in the experiments. The chips are sorted by target azimuth: odd chips are used for training and even chips are used for testing. The peak detection module and the segmentation module described in [1] are applied to the 600 chips to obtain 20 by 40 subimages or regions of interest (ROI) containing the targets. For chips that the segmentation module fails to find the ROI, we segment the chip manually. For each ROI, we also extract a 4 by 40 subimage that contains only the longer leading surface (LLS). LLS is defined as the longer side of the vehicle facing the radar. See Figure 1.
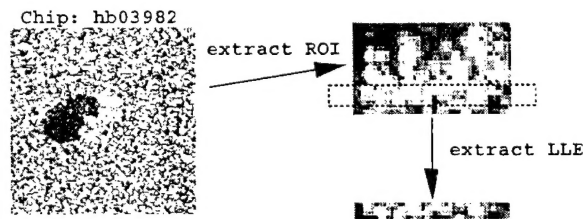
Figure 1: ROI and LLS Subimages Extracted from a Chip for Training and Testing

In the experiment using K-L decomposition, a covariance matrix is computed from the training images. The 10 eigenvectors of the matrix with the largest eigenvalues constitute the feature space axes. When a test image is presented to the classifier, the estimated target azimuth, α, is used to select K training images (from each target class) that have the closest azimuth to α to construct a template. Also, K training images (from each target class) that have the closest azimuth to $\alpha \pm 180^o$ are selected to construct an 180° alternative template. In other words, there are $2N$ candidate templates for every test image to compare with, where N is the number of candidate target classes. In the preliminary result presented below, we use $K = 3$ and $N = 3$.

To classify a test image, the test image and the $2N$ templates are projected onto the feature space and the template that has the smallest Euclidean distance to the test image determines the class tag and the pose of the test image. In addition to using the whole ROI to train and test the classifier, we also use the LLS subimages for training and testing. The reason for using LLS subimages is that they are likely to be configuration independent. Confusion matrices are shown in Table 1 and Table 2. Note that our experiment differs from other similar experiments [2] in two aspects. First, we use segmented ROIs. And second, we dynamically construct azimuth-dependent templates for classification.

1

Azimuth-dependent templates can account for the great variability of SAR imagery and, therefore, improve recognition rate.

Table 1: Recognition Using ROIs

	BTR70	BMP2	T72	Other
BTR70	84	2	2	12
BMP2	1	79	7	13
T72	1	7	69	23

Table 2: Recognition Using LLS

	BTR70	BMP2	T72	Other
BTR70	69	9	9	16
BMP2	9	65	15	11
T72	5	12	69	14

Recognition based on cross correlation shows comparable results as K-L-based recognition (Table 3). As in the case of K-L-based recognition. each test image will have six templates to compare with. For each template we use leading surfaces and peaks to find possible alignment of the test image and the template as in Section . The alignment that gives the best cross correlation will be the score of the template. and the template with the highest score determines class tag and pose of the test target. Cross correlation of two images, f and g, is computed using the following expression:

$$Cross\ correlation = \frac{\sum_{m,n}(f_{m,n} - \overline{f}) \cdot (g_{m,n} - \overline{g})}{\sigma_f \cdot \sigma_g}. \tag{1}$$

Table 3: Recognition Using Cross Correlation

	BTR70	BMP2	T72	Other
BTR70	80	2	5	13
BMP2	0	73	13	14
T72	0	4	88	8

Observed length and width of a ROI can also be used for recognition, assuming no or little occlusion. For most target classes, recognition using length and width is independent of configuration changes on the deck of a target. We construct from the training images piecewise linear functions to approximate the length and width of a target as target

2

azimuth varies. An example is shown in Figure 2. The product of Gaussian likelihoods is used as a score for classification. In expression (2), l_o and w_o are the observed length and width of the ROI, respectively; $l_{i,\alpha}$ and $w_{i,\alpha}$ are the stored length and width of the ith target class at azimuth $= \alpha$. Table 4 shows the confusion matrix of our experiment. Note that $\sigma = 1$ pixel is used.

Likelihood

$$= p(l_o, w_o \mid l_{i,\alpha}, w_{i,\alpha}) = p(l_o \mid l_{i,\alpha}) \cdot p(w_o \mid w_{i,\alpha}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(l_o - l_{i,\alpha})^2}{2\sigma^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(w_o - w_{i,\alpha})^2}{2\sigma^2}} \quad (2)$$
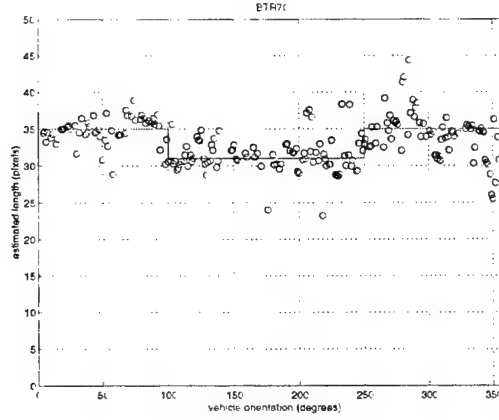

Figure 2: A Piecewise Linear Function Constructed from the Training Images to Approximate the Length of the BTR70 at Different Azimuth

Table 4: Recognition Using Length and Width

	BTR70	BMP2	T72
BTR70	77	19	3
BMP2	7	89	15
T72	5	22	80

3

2. Structural Model-Based Recognition Approach

Statistical classification, view-based recognition, and model-based recognition are the techniques commonly seen in the computer vision literature. A structural model-based approach is adopted in this research effort.

Because of the great variability of SAR imagery. orientation-dependent templates o1 models are often used for recognition. regardless of which technique is adopted. Section describes how a orientation-dependent model is selected in our approach.

Statistical classification is computationally tractable. However, our early experiments with statistical classification (K-L decomposition; eigenimage approach) [2] achieved only about an 80 percent success rate in recognition.

The view-based approach matches the input image to every possible view of all target classes. It has two fundamental limitations: 1) It lacks a good indexing capability that selects a few candidate target classes from the entire database for matching, and 2) It is computationally prohibitive to maintain a set of views that covers a target under the extended operating conditions (EOCs). Take the T72 tank, for example. With the hatch open or not, oil tanks or not will result in different images [3]. In other words, recognition with the view-based approach becomes a combinatorial problem.

There are a few traits of a model-based recognition system including low level (image level) feature detection/extraction, generic hypothesis generation, indexing at various levels of the system, and matching. It is tractable and intuitive to humans to integrate a probability framework, known as Bayesian network, to a model-based system to make decisions at various levels (e.g., image level, component level, and object level) of the system.

2.1 Generic Vehicle Model

We adopt a generic vehicle model unique in the ATR community to define parameters of length, width, and orientation to use in recognition. Observing that most ground vehicles of interest have a rectangular chassis, it is natural to fit a rectangular bounding box to a set of image features to estimate a pair of leading surfaces that in turn give the estimated length, width, and orientation. The rectangular bounding box model can be extended to include articulating parts that accommodate features extruding outside of the box, such as a gun or antenna.

The estimated leading surfaces provide a coordinate frame (called target coordinate frame) local to the target. With the strong assumption that amplitudes, and relative positions of enough peaks (bright points) in the image are quasi-invariant for a small range of target orientation (with respect to the radar) and variations in radar viewing parameters, target coordinate frames can be used to align images for matching individual image features. Typically, 10 to 15 peaks that persist for about 3 to 5 degrees is a sufficient condition for

4
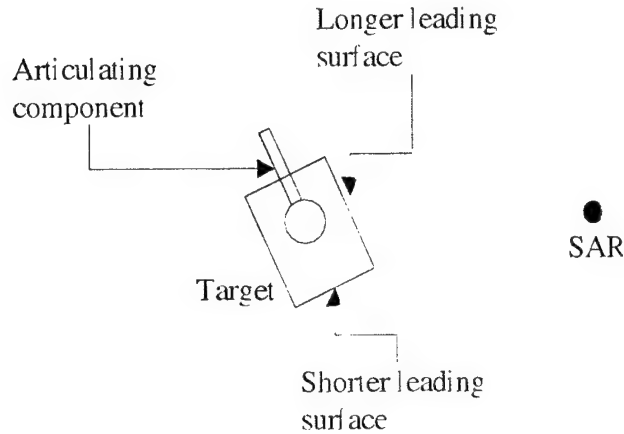
Figure 3: Generic Vehicle Model (Top View)

the quasi-invariance assumption; this condition is verified in this report (Section).

2.2 System Overview

Figure 4 gives an overview of our SAR ATR system. The system has two phases – the hypothesis generation phase that selects a few candidate target classes from the target database, and the hypothesis verification phase that outputs the best class and pose for the test image.

The hypothesis generation phase starts with estimating low-level image features (e.g., peaks), finding clusters of peaks that correspond to a target area. The generic vehicle model is fitted to the target peaks to get target length, width, and orientation estimates. Target peaks and the estimated parameters are used in target indexing.

The hypothesis verification phase aligns candidate class images to the test image, and this process is sped up with the help of the Delaunay walk algorithm. The Evaluation module selects the best class and pose for the test image.

System Overview

Test image   Hypothesis   Candidates   Hypothesis   Class and pose

Generation   Verification
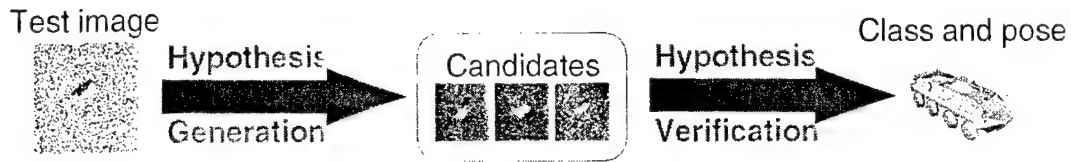
Generation phase:

- Low-level feature estimation.
  e.g., peaks and edgels
- Segmentation/ Extended feature estimation.
  e.g., find clusters of target peaks
- Generic model estimation.
  length, width, and orientation
- Target indexing.
  compute candidates

Verification phase:

- Image alignment.
  find feature correspondences
- Delaunay walk.
  speed up image alignment
- Hypothesis evaluation.
  select the best match from candidates

Figure 4: System Overview

6

3. Persistent Scattering

Because of the great variability, doing recognition with SAR image features, such as peaks (bright points) was once considered impossible. In recent years, researchers have demonstrated that peaks that are persistent over a few degrees of the target orientation (Figure 5) with respect to the radar are sufficient for recognition.
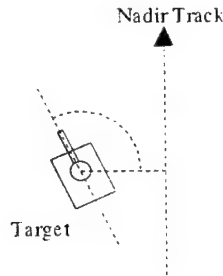

Figure 5: Target Orientation (Top View)

Persistent scattering of inverse SAR (ISAR) and synthesized SAR (XPATCH) imagery were studied by Dudgeon et al. [4] and Binford et al. [5] respectively. In this report, we study persistence of the MSTAR (Moving and Stationary Target Acquisition and Recognition) imagery. The results give evidence that there are enough scatterers with sufficient persistence for structural model-based recognition.

We study persistent scatterers in a set of MSTAR images of three different vehicles (BTR70, BMP2, and T72). The evaluation of persistent scattering was done in three ways: by human observers, by an interactive user interface and a human operator, and by an automated program. A user interface or an automated program is needed because we do not have the ground truth for registering images as in the case of synthesized SAR or ISAR imagery.

Images of targets were examined and compared to the miniature vehicle models we assembled to determine the salient scatterers and the azimuth intervals over which they are visible. For example, one of the BTR70 hatches is visible over the azimuth intervals: 0 to 15, 35 to 50, 85 to 110, and 150 to 175 degrees.

3.1 Peak Detection

Image peaks (bright points) are the primary low-level image features used in our current SAR ATR system. Peaks on the target area are strong radar returns from scatterers on the targets. Corner reflectors, dihedrals, and planes are commonly seen simple scatterers. Different scatterers have different stability; for example, a corner reflector is more stable than a plane scatterer in that energy is more likely to be reflected back from a corner reflector.
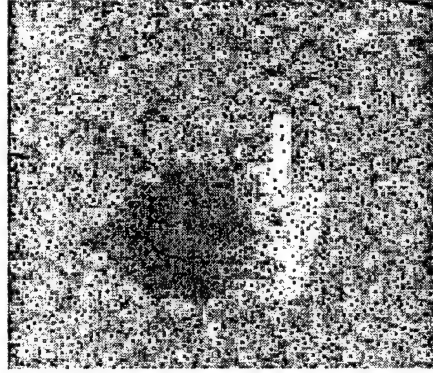
7

Figure 6: Detected Peaks
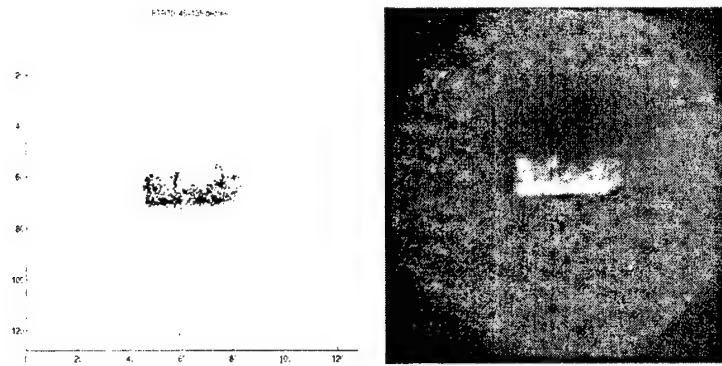
Figure 7: Left: Superposition of Rotated Peaks; Right: Superposition of Rotated Intensity Images

A generic peak detector by Wang and Binford [1] is used to detect image peaks. Figure 6 shows an example of detected peaks overlaid on the original intensity image. Peaks belong to targets are segmented (selected) before persistent scattering is studied.

3.2 Peak Persistence

The peak detector [1] was applied to an image to extract peaks (bright points) in the image. The set of peaks were then rotated to zero azimuth angle and an interactive user interface was subsequently used to register them to a set of reference peaks. This process was repeated for images with target azimuth between 45 and 135 degrees. Figure 7 shows the superposition of the rotated peaks and intensity images of the BTR70.

To automate the registration process, we developed a method based on leading surfaces and ideogram. We use leading surfaces to establish local coordinate frames for the two sets of peaks being registered. We can use a pair of peaks, one peak from each frame, that have the same local coordinates for registration (Figure 8(a)). However, some peaks may not have the corresponding peaks in the other frame due to the great variability of SAR

8

imagery. Therefore, we chose to generate correspondence hypotheses for the 2 frames using the 10 brightest peaks and select hypothesis with largest overlap, where overlap is measured by ideogram (Figure 8(b)). The ideogram counts the number of matched peak pairs and is expressed as follows:
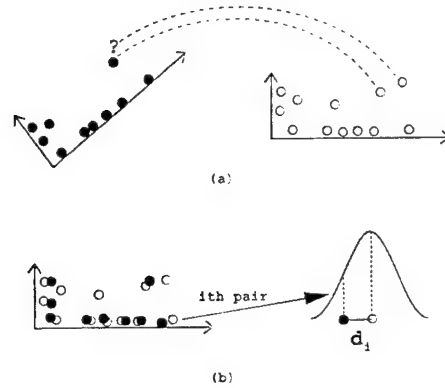
$$Ideogram = \sum_i e^{-\frac{d_i^2}{2c^2}} \tag{3}$$

Figure 8: (a) Local Coordinate Frames Used to Establish Correspondence of Peaks; (b) Ideogram Associated with the ith Peak Pair

Since we have established the correspondence of peaks from one frame to the next, we can generate plots of average number of peaks per frame versus minimum persistence (in degrees) and plots of number of peaks that persist for more than 1, 10, or 20 degrees versus target azimuth (Figure 9). These kinds of plots were introduced by Dudgeon et al. [4] for evaluating persistent scattering in ISAR imagery. The plots show that on the average there are 10 peaks that will persist for more than 15 degrees.

Existence of persistent scattering is the basis for structural model-based target recognition with SAR imagery. Matching of individual scatterers (target features) enables recognition under articulation, obscuration, and configuration changes of targets.

With enhanced image alignment technique, we have improved the ability of our automatic programs to better tracking of target scatters across image frames. Figure 10 (a) shows the superposition of two aligned peak images. Figure 10 (b) shows the superposition of 21 peak images, all registered to the 11th peak image. Note that the effect of range foreshortening is corrected and all of the peak images are rotated to zero azimuth.

We use a set of 231 BTR70 images in our study of persistent scattering with automated programs. The result shows that on the average there are 10 peaks that persist for more

(a) Average number of peaks per frame versus minimum persistence

(b) Number of peaks that persist for more than 1, 10, or 20 degrees versus target azimuth
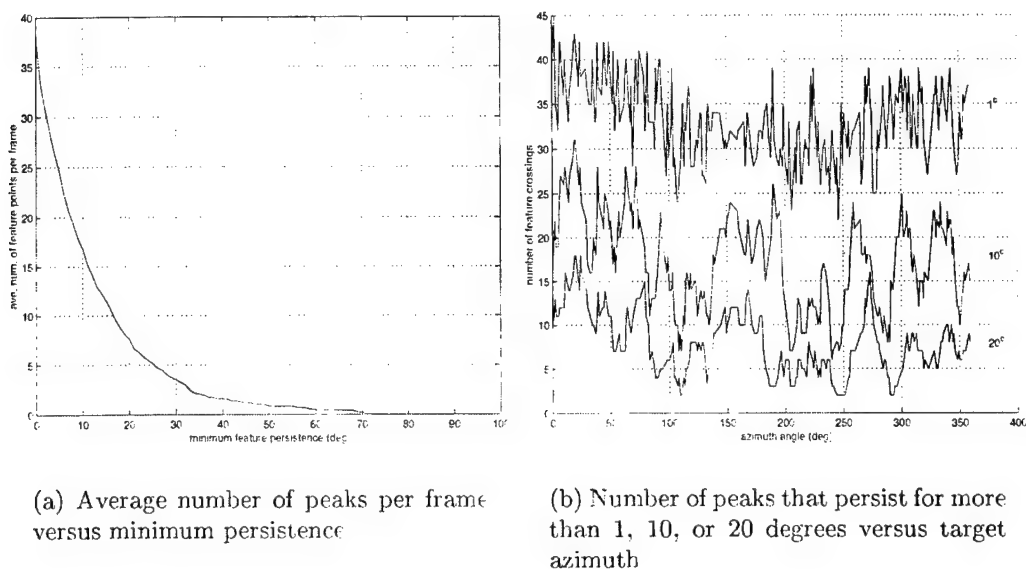
Figure 9: Persistent Scattering

(a) superposition of 2 BTR70 peak images

(b) superposition of 21 BTR70 peak images
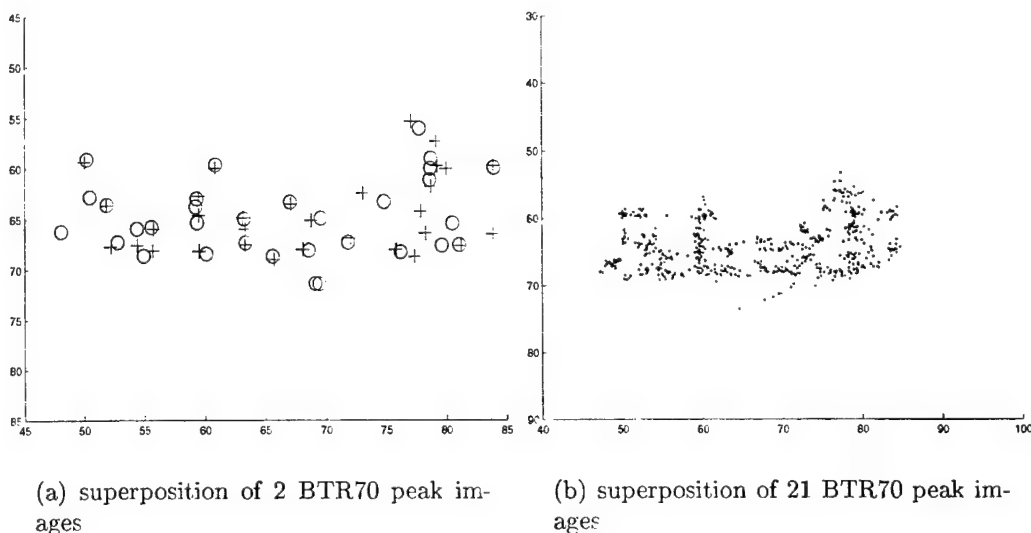
Figure 10: Alignment of Peak Images

than 15 consecutive degrees. This characterization underestimates somewhat the persistence of scatterers that are obscured at some angles and then reappear. Nevertheless, it gives evidence that there are enough scatterers with sufficient persistence for structural model-based recognition.

(a) A tophat-like scatterer   (b) A dihedral-like scatterer
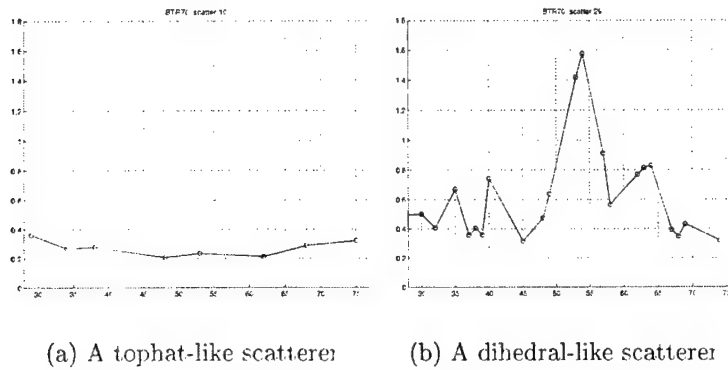
Figure 11: Amplitude Variation of Peaks

We also studied the variation of peak amplitude as a function of azimuth angle (Figure 11). The preliminary result shows that it may be possible to classify target peaks into a handful of categories. Knowing the category of a peak allows us to relate it back to the physical component of the target that reflected the radar energy. Also, statistics of appearance of each peak category can be collected for a given target from the training imagery.

4. Target Segmentation

We use MSTAR target chips for the development and testing of our ATR algorithms. We segment out targets before performing estimation of parameters (such as length, width, and orientation) of a generic vehicle model.

The segmentation module is a reimplementation of the segmentation technique developed by Wang and Binford [1] for SAR imagery. The technique involves peak detection, peak selection, Delaunay triangulation. and breaking long links in the triangulations. Failures of peak detection and target segmentation were analyzed. Plans have been made for improving performance.

The peak detector developed by Wang and Binford is used to estimate position, amplitude, and widths of strong radar returns in the input image. Thresholds for peak amplitude are set to select strong peaks possibly corresponding to target scatterers since amplitude of the returned radar signal is probably the best quantity for discriminating targets against clutter. The Delaunay triangulation of the strong peaks is computed. Long links in the triangulation are broken to segment out target areas because the density of strong peaks are low in the nontarget area.

(a) Intensity image of a BTR70

(b) Strong peaks (target) and their triangulation

(c) Weak peaks (shadow) and their triangulation

(d) Segmented target and shadow areas
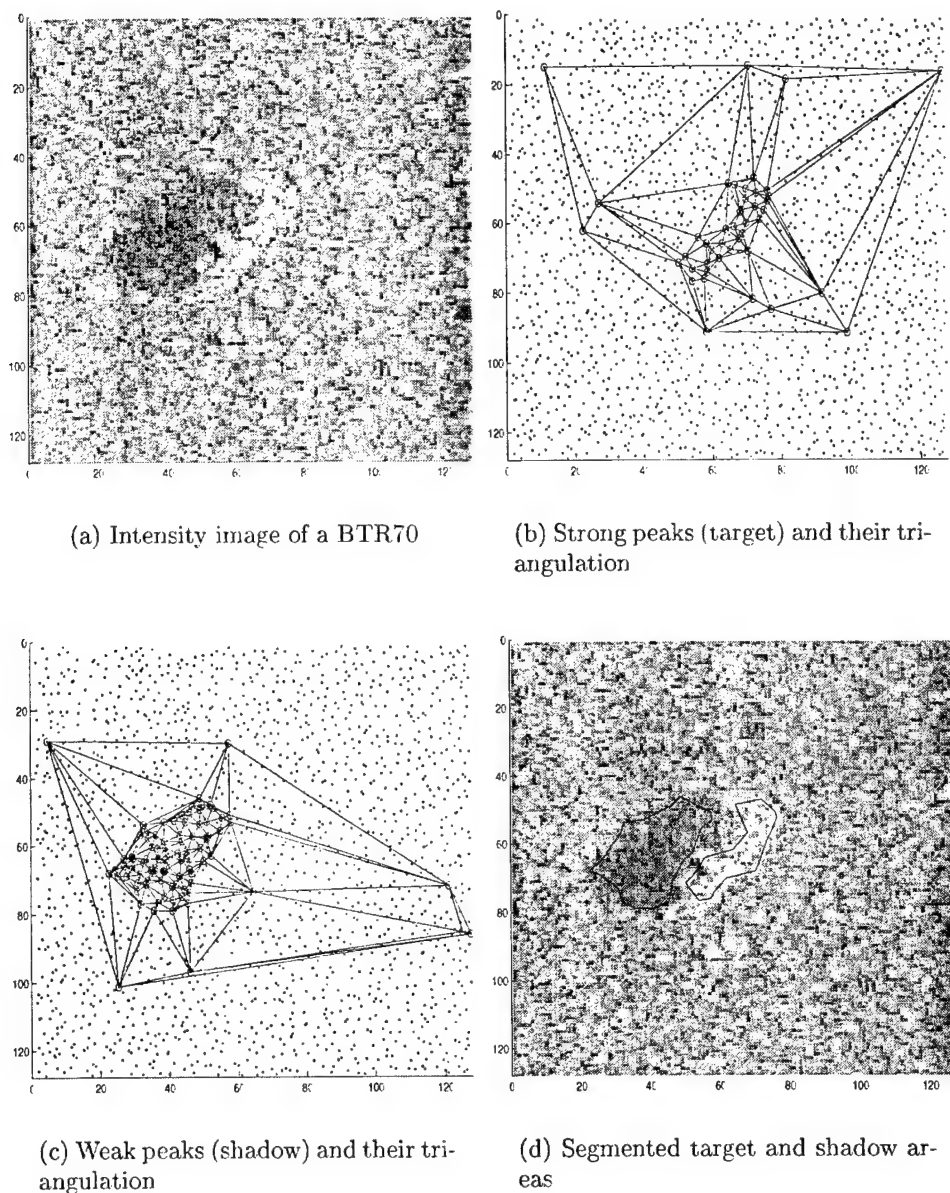
Figure 12: Segmentation

Figure 12 (a) shows the input intensity image of a BTR70. The peak detector [1] is applied to extract peaks (bright points) in the image. Figure 12 (b) shows the Delaunay triangulation of strong peaks with amplitude greater than 2.0. These peaks correspond to the class of target scatterers. After breaking long links (> 10 pixels) in the Delaunay triangulation, a few groups of connected strong peaks are formed. These groups can be further examined and ranked using the prior knowledge of average peak amplitude and target sizes. Other types of objects can be segmented in the same way. For example, Figure 12 (c) shows the Delaunay triangulation of the weak peaks that possibly correspond

13

to shadows. Figure 12 (d) demonstrates segmented target and shadow areas.

5. Higher Order Features

To go beyond using point features (e.g., image peaks) for indexing and matching, higher order features such as pairs of peaks, line features. and boundaries can be used. This section summarizes our preliminary results on finding line features and shadow boundaries. Line features are observable in almost all ground targets. for example, the leading surfaces, the gun of a T72, and the shadow of the gun. Shadow boundaries contain information not available from the target area. for example, the height of the target, and the aim angle of the gun. A more detailed analysis should also take into account the effect of layover.

The Binford-Chiang edge operator is first applied to extract edgels (edge elements) and these edgels are linked into higher order features. Delaunay triangulation is used as a means of spatial indexing that establishes a neighborhood for each edgel; in other words, the neighboring edgels in the Delaunay triangulation are candidates for linking.

5.1 Line Features

Figure 13 shows a T72 image. Figures 14(a) and 14(b) show the detected delta edgels and step edgels, respectively. The example here focuses on linking step edgels.

First, edgels are detected using the Binford-Chiang edge operator. To link the edgels, direct neighbors in the Delaunay triangulation (Figure 14(c)) are used as candidates. The best candidate is selected based on a probability measure. and this process is continued until the probability is smaller than a predetermined threshold. Figure 14(d) shows the line features with at least two step edgels. As can be seen. step edges that correspond to the gun, shadow of the gun. and part of the leading surfaces are good line features.
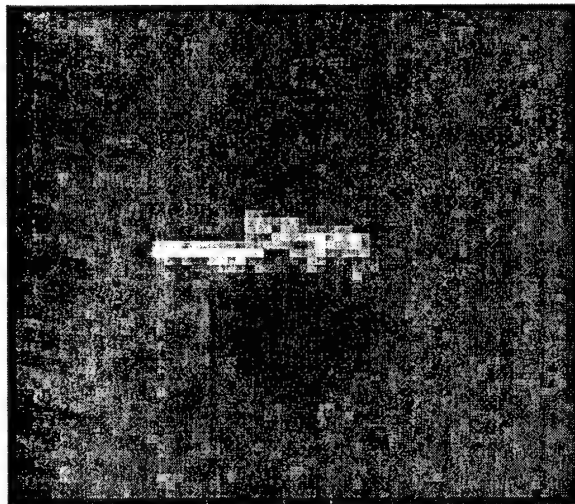

Figure 13: T72 Image

15

(a) Delta edgels

(b) Step edgels

(c) Delaunay triangulation of step edgels
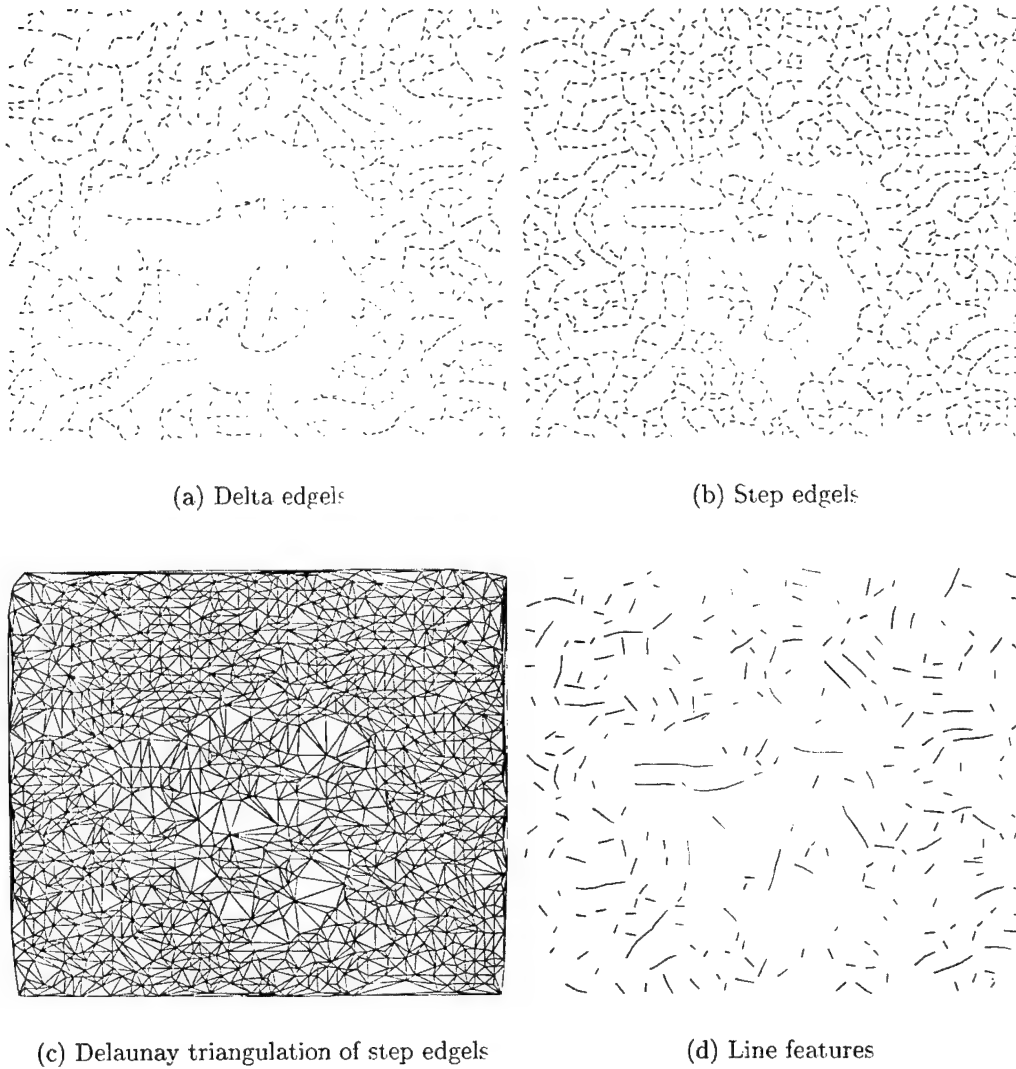
(d) Line features

Figure 14: Finding Line Features

To assess the goodness of fit, the line defined by the first edgel and the last edgel is used as a reference line. The deviation of the tangent of each edgel from the reference line is modeled by a normal distribution with zero mean, and standard deviation σ. In other words, $t_i \sim N(0, \sigma)$ (Figure 15). To allow more expected variation with a larger number of edgels on the line, Chi-square distribution is used to evaluate the overall goodness of fit of the edgels to the line. $\sum_{i=1}^{n} (\frac{t_i}{\sigma})^2 \sim \chi^2$ with degrees of freedom = n. Other edgel features such as amplitude (intensity), contrast, and curvature can be incorporated to improve decision.

Figure 15: Probability Model

5.2 Shadow Boundaries

A maximum likelihood decision method by Oliver et al. [6] is adopted to select candidate edgels that are likely to be on the shadow-background boundaries from the detected step edgels. Then, a hierarchical linking strategy is used to link the candidate edgels into a boundary curve. Delaunay triangulation is used as a means for searching and optimization in linking edgels.

5.2.1 Likelihoods

The method by Oliver et al. [6] uses the ML decision procedure to select one of the two hypotheses: {one distribution, two distributions}. In other words, the pixels on both sides of the edgel have the same or different distribution.

Gamma distribution for L-look SAR: $\Pr(I) = (\frac{L}{\mu})^L I^{L-1} \frac{1}{\Gamma(L)} e^{-\frac{LI}{\mu}}$

In our case (one-look): $\Pr(I) = \frac{1}{\mu} e^{-\frac{I}{\mu}}$, where μ is mean of the distribution.

Take n pixels from each side of the edgel and compute the likelihood of the two hypotheses:

- Two distributions (two distributions characterized by $\mu1$ and $\mu2$):

$$Pr(\mu1, \mu2 \mid data) \sim \prod_{i=1}^{n} \frac{1}{\mu1} e^{-\frac{I_i}{\mu1}} \cdot \prod_{j=1}^{n} \frac{1}{\mu2} e^{-\frac{I_j}{\mu2}} \tag{4}$$

The log-likelihood for two distributions is as follows:

$$\lambda^2(\mu1, \mu2) \sim -n[\ln(\mu1) + \frac{\overline{I_1}}{\mu1} + \ln(\mu2) + \frac{\overline{I_2}}{\mu2}] \tag{5}$$

where $I_1 = \frac{1}{n} \sum_{i=1}^{n} I_i$, $I_2 = \frac{1}{n} \sum_{j=1}^{n} I_j$

- One distribution (characterized by $\mu1$):

$$Pr(\mu1 \mid data) \sim \prod_{i=1}^{2n} \frac{1}{\mu1} e^{-\frac{I_i}{\mu1}} \tag{6}$$

17

The log-likelihood for one distribution is as follows:

$$\lambda^1(\mu 1) \sim -2n[\ln(\mu 1) + \frac{\overline{I_1}}{\mu 1}]$$ (7)

where $I_1 = \frac{1}{2n} \sum_{i=1}^{2n} I_i$

5.2.2 Computing Λ^2 and Λ^1

Oliver et al. [6] use $\overline{I_1}$ and $\overline{I_2}$ as approximations for $\mu 1$ and $\mu 2$. Although they seem to be reasonable approximations, the resulting ML decision rule have very little discriminating power. The approximations favor the one-distribution hypothesis and becomes ineffective with low SNR typical of SAR images; in our case, $SNR = \frac{\overline{S} - \overline{N}}{\sqrt{\sigma_S^2 + \sigma_N^2}} \approx 1$, where 'S' = background and 'N' = shadow.

In our implementation, $\mu 1$ and $\mu 2$ from precollected statistics are used, and we compare the following two quantities to determine which hypothesis is more likely:

$$\Lambda^2 = max(\lambda^2(\mu 1, \mu 2), \lambda^2(\mu 2, \mu 1))$$ (8)

$$\Lambda^1 = max(\lambda^1(\mu 1), \lambda^1(\mu 2))$$ (9)

i.e.,

- hypothesis = one distribution, if $\Lambda^1 > \Lambda^2$.

- hypothesis = two distributions, if $\Lambda^2 > \Lambda^1$.

It is noted that Oliver et al. [6] find edgel location/orientation by maximizing Λ^2 while we use Binford-Chiang edgel operator for edgel detection.

5.2.3 Hierarchical Linking

Two Delaunay triangulations are used in the linking process (Figure 17): DT^a, Delaunay triangulation of all edgels, and DT^b, Delaunay triangulation of shadow-background boundary edgels (selected using ML decision).

In the first linking stage, shadow-background boundary edgels that are direct neighbors in DT^a are connected to form edgel chains. In the second linking stage, end points of edgel

18

Figure 16: Shadow-Background Boundary Edgels

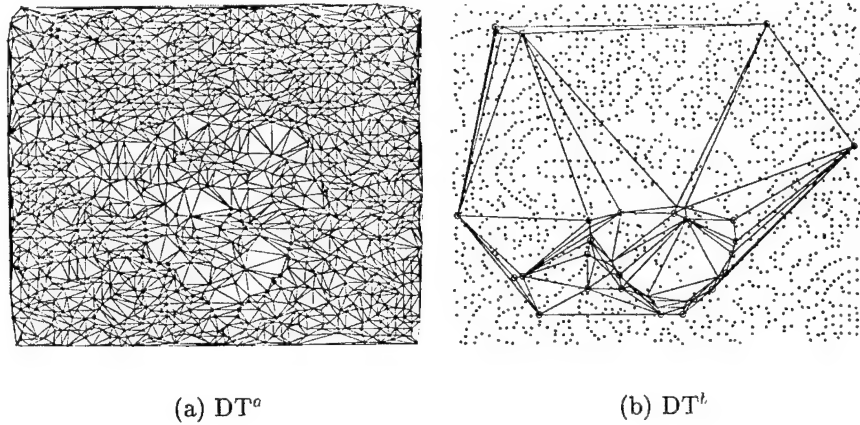(a) DT^a (b) DT^b

Figure 17: Delaunay Triangulations

chains that are direct neighbors in DT^b can be linked to form a boundary curve. Figure 18 shows the results of stage #1 and stage #2 linking.

Figure 19 shows a detailed example of linking edgel chains. There are three direct neighbors for the edgel p in DT^b (Figure 19(a)). The Delaunay walk in DT^a is used to find initial paths from p to the three neighbors in DT^b. These initial paths can be optimized with respect to contrast or a probability quantity to get the paths shown in Figure 19(b), and the best path can be selected using, for example, contrast.

(a) Edgel chains
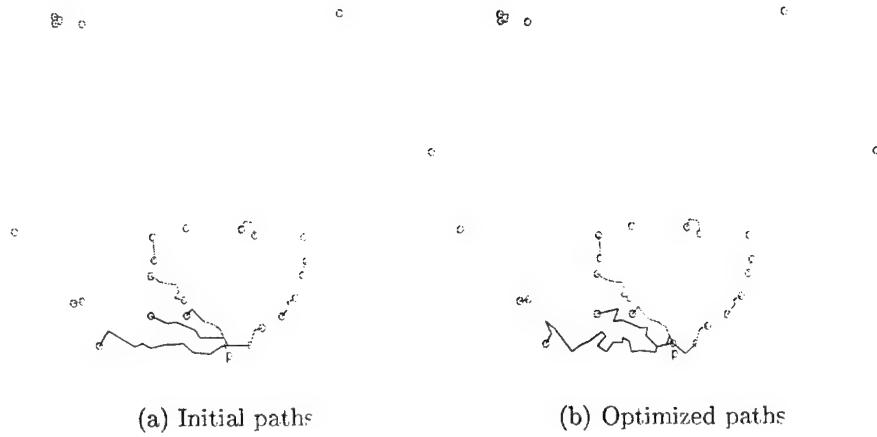
(b) A boundary curve

Figure 18: Hierarchical Linking

(a) Initial paths

(b) Optimized paths

Figure 19: Hierarchical Linking

6. Leading Surface Estimation

We define leading surfaces to be the sides of the vehicle that face the radar. The LLS usually corresponds to one of the two sides of the target and the shorter leading surface (SLS) corresponds to either the front or the end of the target. Figure 20 shows a top view illustration. The LLS determines target azimuth up to a 180-degree flip. The role of LLS and SLS may be changed in the case of occlusion because the sides may appear shorter than the end (or the front).

Leading surfaces can be used in several subsystems in our ATR framework. Also, they can be used to reduce computational cost and/or improve recognition rate in different recognition subsystems, e.g., statistical classification, template-based, and model-based.
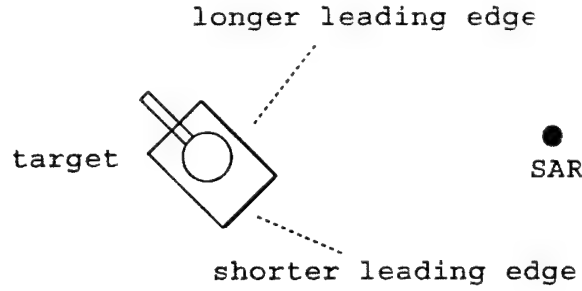

Figure 20: Leading Surfaces, the Sides of the Vehicle Facing the Radar

6.1 March 1998

We combine amplitude-weighted ideogram and a probability-like weighting strategy to find the best leading surface (Figure 21 (a)) as shown in expression (10). Amplitude-weighted ideogram finds surfaces with multiple peaks close to them, and probability-like weighting rejects surfaces that run across the body of the target. Figure 21 (b) shows examples of good and false leading surfaces.

$$\underbrace{\left(\sum_i A_i e^{-\frac{d_i^2}{2\sigma^2}}\right)}_{amplitude-weighted\ ideogram} \cdot \underbrace{\left(\prod_{i:d_i>0}\left[1 - \frac{A_i}{A_{max}}(1 - e^{-\frac{d_i^2}{2\sigma^2}})\right]\right)}_{probability-like\ weighting} \tag{10}$$

The initial result of our leading surface estimator exhibited a sinusoidal bias, as shown in Figure 22. There are two causes of this sinusoidal bias: the first cause is inherent from the estimation algorithm, and the second one comes from SAR imaging geometry.
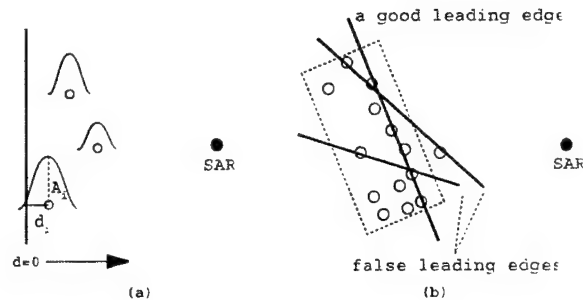
21

Figure 21: (a) Example of a Leading Surface and Ideogram. and (b) Examples of Leading Surfaces
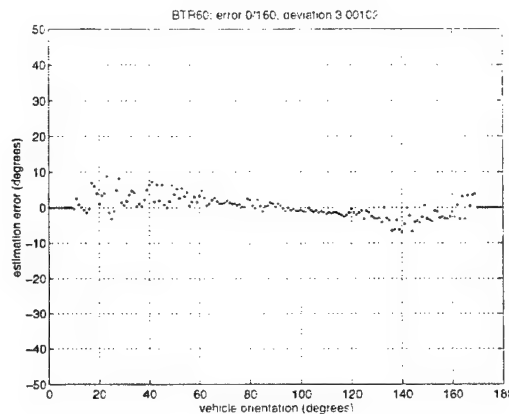

Figure 22: Sinusoidal Estimation Bias

Because we only tried to estimate the LLS, peaks on the SLS will pull the estimated LLS toward the SLS, as shown in Figure 23. Our solution is to estimate the two leading surfaces, LLS and SLS, together. Peaks are assigned to either the LLS or the SLS before we compute ideogram.

Since SAR images are the projection of targets onto the slant plane, the dimension in the range direction will be foreshortened by a factor of $cos\phi$, as shown in Figure 24. As a result, the angle of an surface will be increased or decreased slightly. To remedy this problem, simple geometric correction is applied to compensate for the foreshortening before leading surface estimation is performed.

Table 5 summarizes the current state of performance of our target azimuth estimation using leading surfaces. We characterize the performance using two numbers; error probability, P_e, is the number of estimation mistakes normalized by the total number of estimations, and the root-mean-square, RMS, error measures the fluctuation of the correct estimations. We define an estimation with an error less than 10 degrees to be correct. The average RMS error is less than 3 degrees, which is slightly better than human observers.
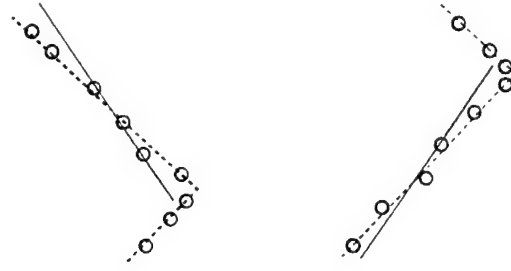
22

Figure 23: True Leading Surfaces (Dashed Lines) and Estimated Leading Surfaces (Solid Lines)
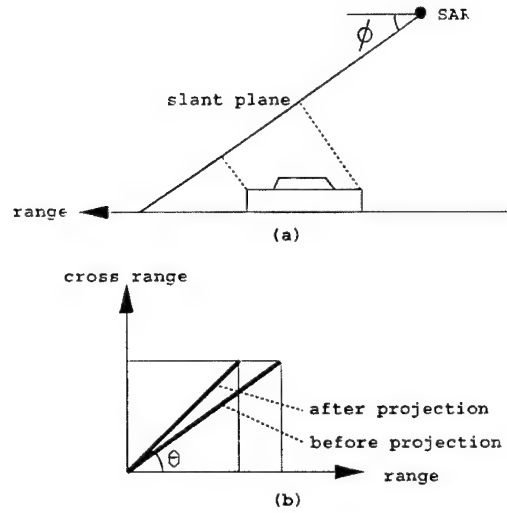
Figure 24: (a) SAR Imaging Geometry and Foreshortening (Side View), and (b) LLS angle before and after Projection

Table 5: Performance of LS Estimation

	RMS error	P_e
BTR70	2.5^o	3/168
BMP2	2.8^o	2/231
T72	3.0^o	4/203

6.2 November 1998

We have made progress in leading surface estimation. Table 6 shows our previous result on leading surface estimation, and Table 7 shows the new result.

The progress is a result of a few algorithmic improvements. First, LLS hypotheses with

Table 6: Performance of LS Estimation

	RMS error	P_ϵ
BTR70	2.47^o	$8/233 \approx 3.43\%$
BMP2	2.79^o	$5/231 \approx 2.16\%$
T72	3.10^o	$7/231 \approx 3.29\%$

Table 7: Performance of LS Estimation

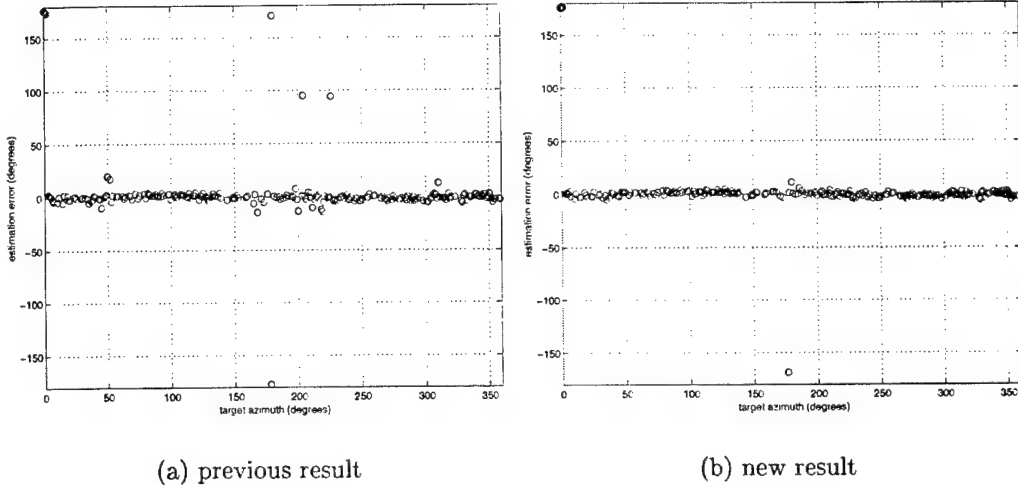|	RMS error	P_ϵ
BTR70	2.20^o	$2/233 \approx 0.86\%$
BMP2	2.66^o	$1/231 \approx 0.43\%$
T72	3.28^o	$6/231 \approx 2.60\%$

(a) previous result   (b) new result

Figure 25: Orientation Estimation Errors of a BTR70

longer length are emphasized because they are less sensitive to peak position fluctuations (note that LLS hypotheses are generated from peaks). Second, the relation between projected length and width (i.e., length > width) is used to eliminate most 90-degree estimation errors (Figure 25). Third, we use a true probability instead of a probability-like weighting for leading surfaces; i.e., we use the following metric to rank leading surface hypotheses:

$$\underbrace{\left(\prod_{i:d_i>0} \left[1 - Pr^t_i \cdot Pr^o_i \right] \right)}_{probability} \cdot \underbrace{\left(\sum_i A_i e^{-\frac{d_i^2}{2\sigma^2}} \right)}_{ideogram} \tag{11}$$

24

where $Pr^t{}_i = $ Pr(the ith peak is a target peak) and $Pr^o{}_i = $ Pr(the ith peak is outside the leading surface).

6.3 June 1999

The following equation yields a probability measure for leading:

$Pr(leading\ edge)$

$$\approx \prod_{outside\ peaks} [1 - Pr(ith\ peak\ is\ a\ target\ peak) \cdot Pr(ith\ peak\ is\ outside)] \tag{12}$$

where:
$Pr(ith\ peak\ is\ a\ target\ peak) = \int_0^{a_i} Rayleigh\ dx;\ \uparrow\ as\ a_i \uparrow$
$Pr(ith\ peak\ is\ outside) = \int_0^{n_i} \frac{2}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}\ dx;\ \uparrow\ as\ n_i \uparrow$
Compensate according to the number of peaks outside: $[Pr(leading\ edge)]^{\frac{1}{\#outside\ peaks}}$
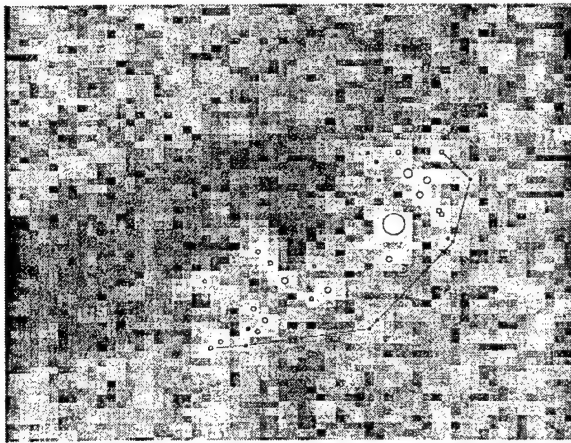
6.4 Alternative Leading Surface Hypotheses

Leading surfaces define a rectangular box which gives target length, width, and orientation estimates independent of target class. In our previous results (Table 8), orientation estimation errors greater than 10 degrees are encountered 1.3 percent of the time (i.e., P_e = 1.3 percent) and the RMS error of the correct estimations is less than 3 degrees (i.e., slightly superhuman). In order to address this 1.3 percent estimation error, we generate alternative leading surface hypotheses by dropping peaks on the front convex hull of the target peaks, where the front convex hull is the half convex hull facing the radar.

Table 8: Performance Achieved by Selecting Single Hypotheses

	RMS error	P_ϵ
BTR70	2.20^o	$2/233 \approx 0.86\%$
BMP2	2.66^o	$1/231 \approx 0.43\%$
T72	3.28^o	$6/231 \approx 2.60\%$

Figure 26(b) shows the leading surface hypotheses generated from the front convex hull shown in Figure 26(a). Because the front convex hull can be inaccurate due to nontarget peaks outside the leading surfaces (at least one nontarget peak will be encountered 20

(a) Front Convex Hull
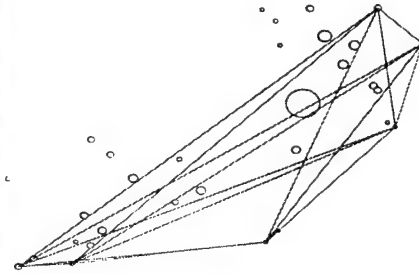
(b) Hypotheses

(c) Front Convex Hull
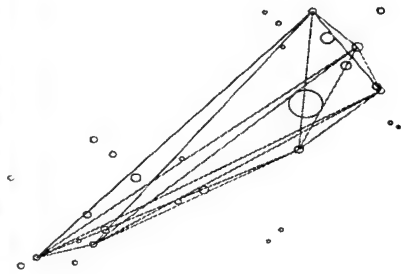
(d) Hypotheses

Figure 26: Alternative Leading Surface Hypotheses

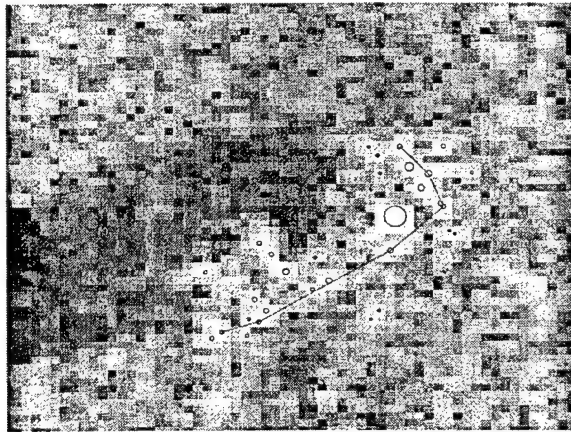percent of the time), peaks on the front convex hull are dropped to get a new front convex hull and a new set of leading surface hypotheses, as shown in Figure 26(c) and (d), respectively. Linear extension of pairs of peaks (from the front convex hull) in the Delaunay triangulation of all target peaks are used as leading surface hypotheses. Notice that peaks dropped are still used for the evaluation of leading surface hypotheses. Also, by using leading surface hypotheses generated from the front convex hulls, a 10 to 1 reduction in hypotheses number (compared to using all pairs of peaks) is achieved.

7. Target Indexing

The target indexing module selects from the target database a few candidate classes for matching; in other words, exhaustive matching is avoided with target indexing. There are two important requirements that the target indexing module must satisfy. First, the candidates must include the correct target classes. Second, this indexing step must be quick.

7.1 The Target Database

Each target class has 100 images in the target database: these images span the 360 degrees evenly. Peak detection, segmentation, and leading surface estimation are applied to each database image to obtain image features and parameters.

All of the database images have an 17 degrees depression angle. For each target azimuth (orientation), a pointer table is created for all target classes. Pointer tables are explained next.

Figure 27: Target Database

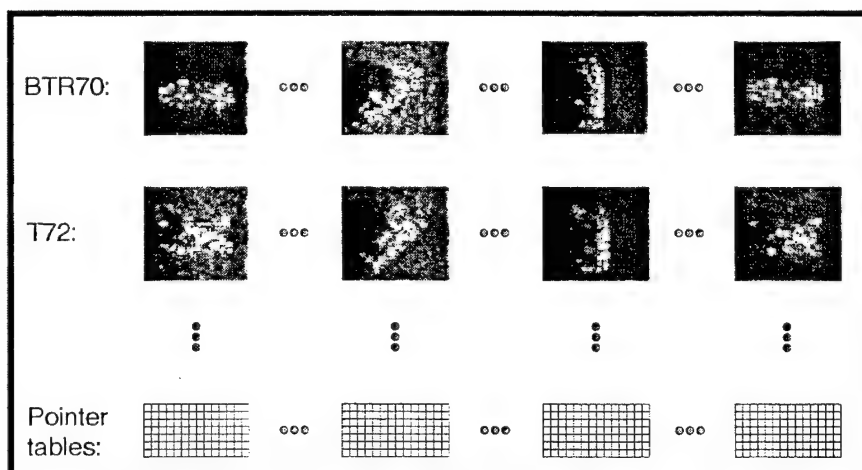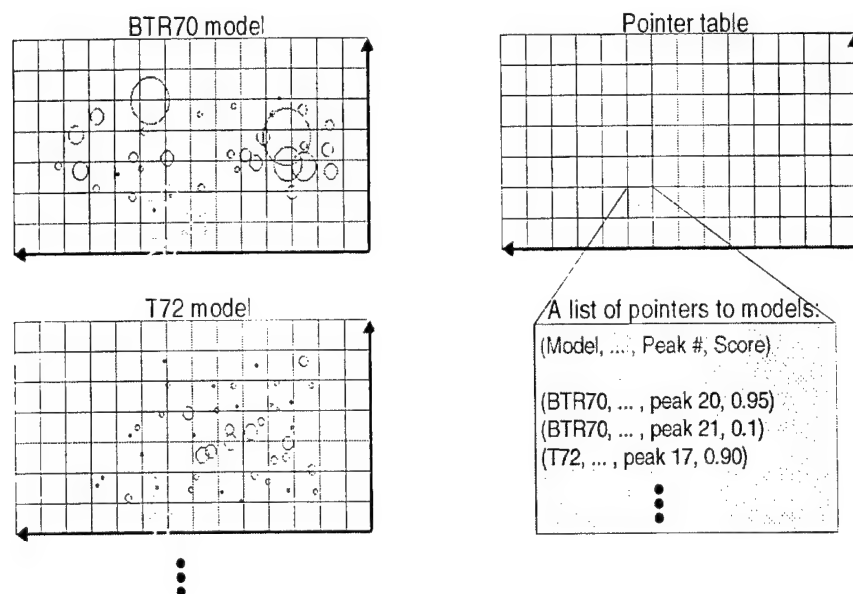## 7.2 Target Indexing

Pointer tables are used for target indexing. Each entry in a pointer table contains a list of pointers pointing to peaks of target classes. For example, the shaded cell in Figure 28(a) has a pointer to peak 20 of the class BTR70 with a score of 0.95. Also, it has another pointer to peak 17 of the class T72 with a score of 0.90. The score can be a probability quantity.
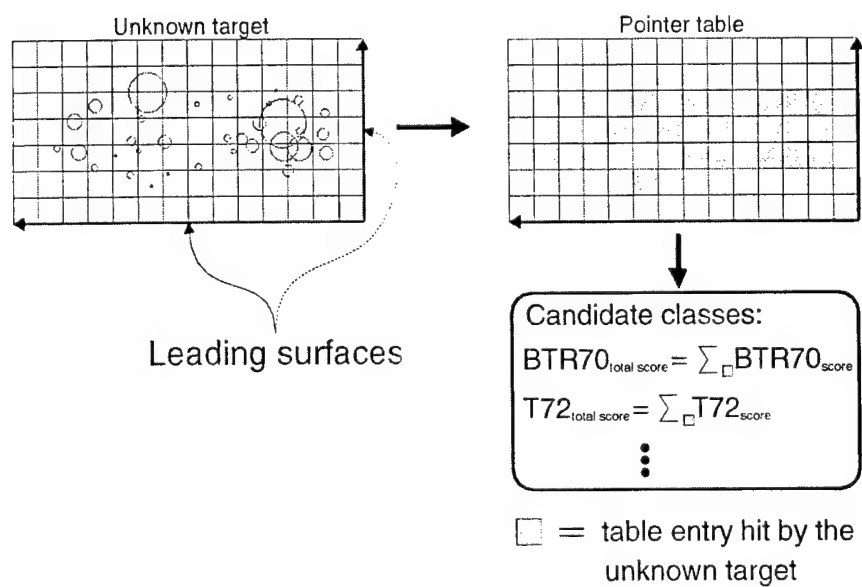
27

When an unknown target is encountered, the table entries hit by the peaks of the unknown target are used for sorting out candidate target classes quickly (Figure 28(b)). For example, the total score for the BTR70 class can be computed by summing all of the scores over all table entries hit by the unknown target. The same procedure is used for other target classes. This step is quick because it involves only a table lookup and addition of scores. More importantly, the indexing procedure uses positive evidence for finding candidate classes. This can be useful under the EOCs.

It is worth pointing out that the axes used here are the leading surfaces from the generic vehicle model. Also, higher order target features may be used to help the discrimination of true target classes against false target classes.

Figure 29 illustrates the entire indexing process. With the azimuth (orientation) information from the estimated generic vehicle model, the pointer table with the closest azimuth is selected. And a few candidate target classes are found with the pointer table. Azimuth-dependent pointer tables are necessary to address the high variability of SAR imagery.

BTR70 model

Pointer table

T72 model

A list of pointers to models:

(Model, ..., Peak #, Score)

(BTR70, ... , peak 20, 0.95)
(BTR70, ... , peak 21, 0.1)
(T72, ... , peak 17, 0.90)

(a) Creating pointer tables

Unknown target

Pointer table

Leading surfaces

Candidate classes:

$BTR70_{total\ score} = \sum_{\square} BTR70_{score}$

$T72_{total\ score} = \sum_{\square} T72_{score}$

$\square$ = table entry hit by the unknown target

(b) Table lookup
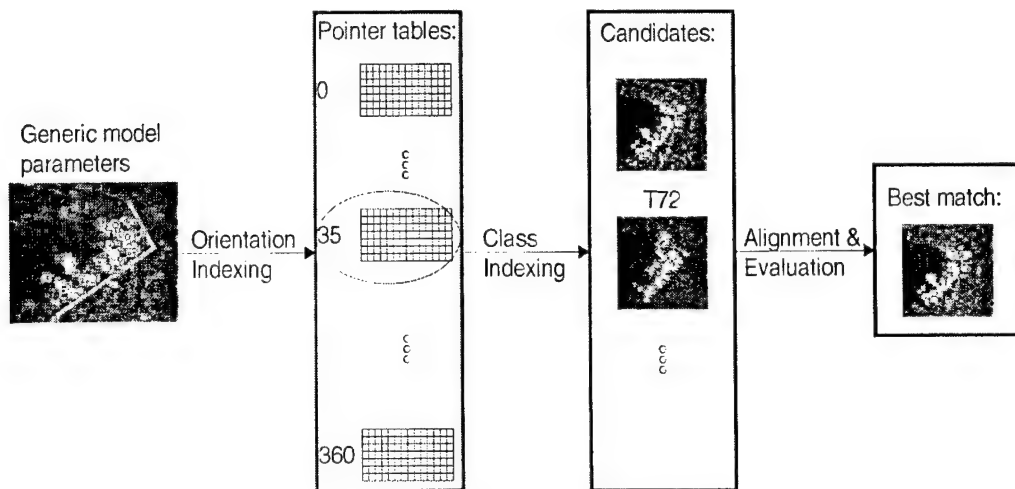
Figure 28: Target Indexing

Figure 29: The Entire Indexing Process

8. Image Alignment

Image alignment is essential in our study of persistent scattering. Also, template images and the input test image must be aligned for recognition. We seek the best rigid body transformation between the input test image and a given template image such that a preselected metric is optimized. Affine transformation is overly general for the application (e.g., it allows shearing of images) and is therefore not used. The alignment procedure has two steps. First, peaks are used to generate initial alignment hypotheses; the hypotheses are ranked by their values of the preselected metric. Second, we refine the best or the first few best initial alignment hypotheses by an analytical formulation. Both hypothesis generation and the analytical formulation aim at efficient image alignment.

8.1 Initial Alignment

Let $S_U = \{(u_x, u_y, A_u)\}$ and $S_V = \{(v_x, v_y, A_v)\}$ be the set of template image peaks and test image peaks respectively. Notice that each peak is characterized by its position and amplitude. We seek a rigid body transformation (T_0) that optimizes a preselected metric (Equation (13)). For aligning peak images, we use $I = \sum_i (A_{ui} + A_{vi}) e^{-\frac{d_i^2}{2\sigma^2}}$ or $I = \sum_i \frac{A_{ui} + A_{vi}}{1 + |A_{ui} - A_{vi}|} e^{-\frac{d_i^2}{2\sigma^2}}$, where $\sigma \sim 1$ pixel is used.

$$
\begin{aligned}
T_0 &= ArgMax_T \ I(S_U, S_V, T) \\
    &= ArgMax_T \ I(S_U, S_V')
\end{aligned}
\tag{13}
$$

where

$$
\begin{aligned}
T &= \begin{bmatrix} \theta \\ a \\ b \end{bmatrix} : \begin{bmatrix} v_x \\ v_y \end{bmatrix} \rightarrow \begin{bmatrix} v_x' \\ v_y' \end{bmatrix} \\
  &= \begin{bmatrix} cos(\theta) & -sin(\theta) & a \\ sin(\theta) & cos(\theta) & b \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ 1 \end{bmatrix}
\end{aligned}
\tag{14}
$$

Initial alignment gives an initial guess of the parameters a and b, i.e., the x- and y-translations. Since the target orientation of the template image is known and the target orientation of the input test image can be estimated from the leading surfaces, the
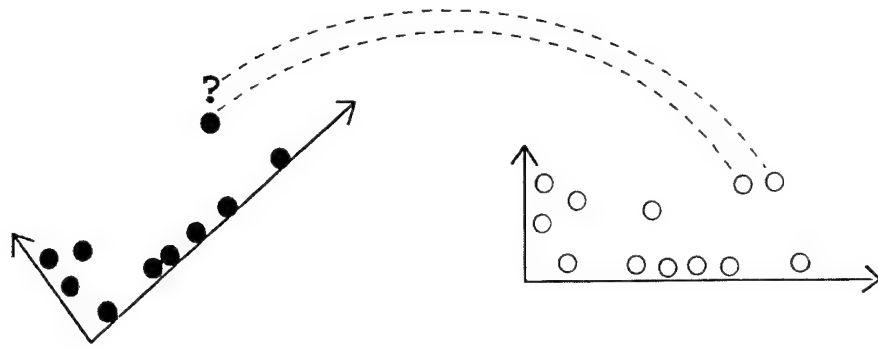
(a) Use peaks for hypothesis generation
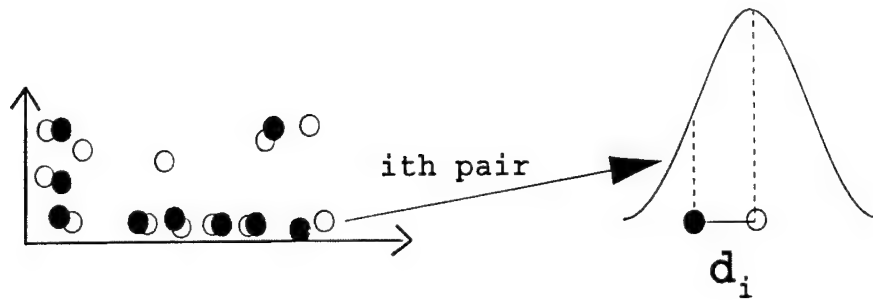
(b) Evaluate hypotheses by overlap of ideogram, $I = \sum_i (A_{ui} + A_{vi})e^{-\frac{d_i^2}{2\sigma^2}}$

Figure 30: Alignment Hypothesis Generation

template image can be rotated such that the two images have the same target orientation, and the best initial guess of θ is zero.

The two leading surfaces together form an intrinsic target coordinate frame that provides quasi-invariant peak coordinates under radar viewing parameter variations. Ideally, two peaks, one from each of the two images being aligned, that have roughly the same coordinates are sufficient for the alignment task. However, some peaks may not have the corresponding peaks due to the great variability of SAR imagery (Figure 30) or false segmentation. Therefore, we choose to generate alignment hypotheses using the five strongest peaks from the input test image and select the best hypotheses with largest overlap, where overlap is measured by weighted-amplitude ideogram defined by $I = \sum_i (A_{ui} + A_{vi})e^{-\frac{d_i^2}{2\sigma^2}}$. Note that strong peaks are preferred because they are more likely to be persistent, and therefore their corresponding peaks in the template images are more likely to be found. Figure 31 shows the initial (translational; $\theta = 0$) alignment hypotheses of two images.

Delaunay walk (Section) is used to enable efficient searches for corresponding peaks. Note that peak pairs have to be formed before $I = \sum_i (A_{ui} + A_{vi})e^{-\frac{d_i^2}{2\sigma^2}}$ can be computed. This is

32

done by local searches in the Delaunay triangulations of the test peaks and the template peaks. The worst case search time is proportional to the diameters of the triangulations.
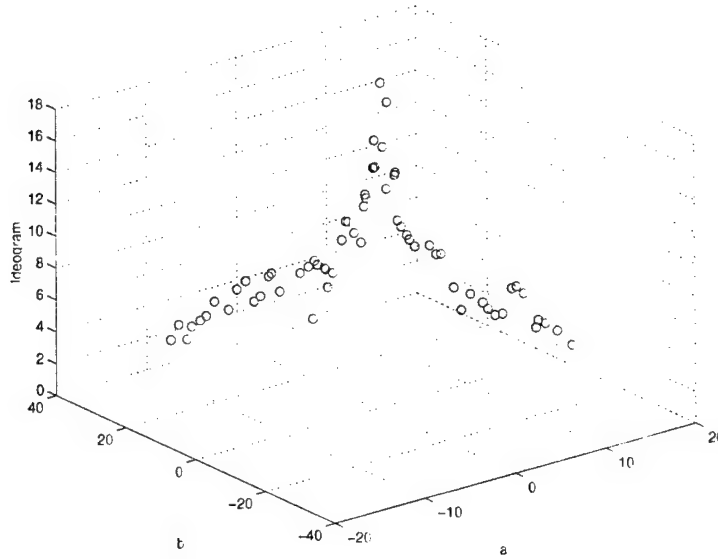

Figure 31: Initial Alignment Hypotheses

8.2 Refined Alignment

Given the sets $S_U = \{(u_x, u_y, A_u)\}$ and $S_V = \{(v_x, v_y, A_v)\}$, I is a function of the transformation T, i.e., I = I(T). The conventional technique for refining alignment is to shift and rotate $S_V = \{(v_x, v_y, A_v)\}$ by a small amount (i.e., sampling) until I(T) reaches a maximum. Instead of using this computationally expensive brute force search, we adopt a Newton-Raphson [7] iterative optimization:

$$
\begin{aligned}
\nabla I(T) &\approx \nabla I(T_0) + H(T_0) \cdot \delta T \\
&= H(T_0) \cdot \delta T \\
&\approx H(T) \cdot \delta T
\end{aligned}
\tag{15}
$$

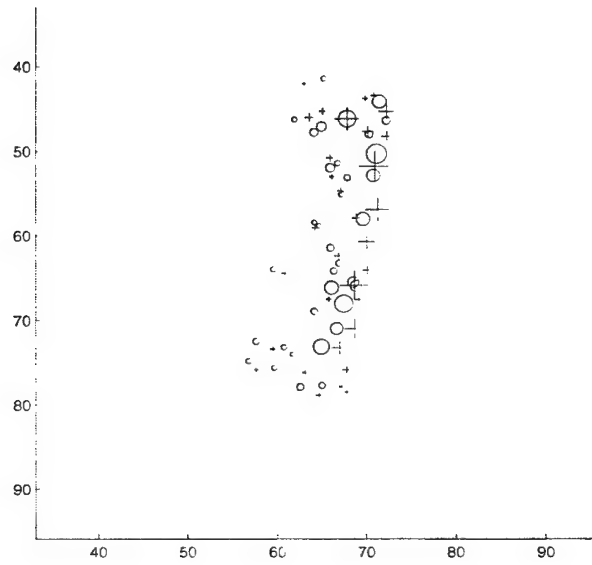where ∇ denotes gradient operator and H is a Hessian matrix. Therefore,

$$
T_0 \approx T - H(T)^{-1} \cdot \nabla I(T)
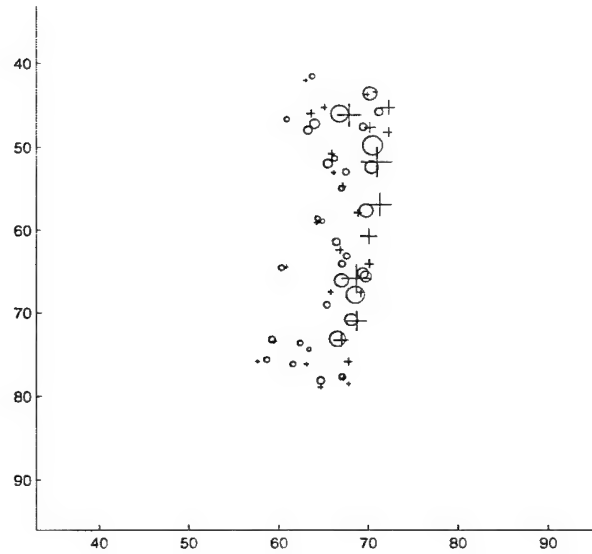\tag{16}
$$

In iterative form,

33

$$T_{i+1} \approx T_i - H(T_i)^{-1} \cdot \nabla I(T_i) \tag{17}$$

In expanded form:

$$\begin{bmatrix} \theta_{i+1} \\ a_{i+1} \\ b_{i+1} \end{bmatrix} \approx \begin{bmatrix} \theta_i \\ a_i \\ b_i \end{bmatrix} - \begin{bmatrix} \frac{\partial^2 I}{\partial \theta^2} & \frac{\partial^2 I}{\partial \theta \partial a} & \frac{\partial^2 I}{\partial \theta \partial b} \\ \frac{\partial^2 I}{\partial \theta \partial a} & \frac{\partial^2 I}{\partial a^2} & \frac{\partial^2 I}{\partial a \partial b} \\ \frac{\partial^2 I}{\partial \theta \partial b} & \frac{\partial^2 I}{\partial a \partial b} & \frac{\partial^2 I}{\partial b^2} \end{bmatrix}_{T_i}^{-1} \begin{bmatrix} \frac{\partial I}{\partial \theta} \\ \frac{\partial I}{\partial a} \\ \frac{\partial I}{\partial b} \end{bmatrix}_{T_i} \tag{18}$$

34

(a) Initial alignment, $\theta = 0^o$, a=-2.8920 pixels, b=-1.9128 pixels

(b) Refined alignment, $\theta = -5.6721^o$, a=-2.3513 pixels, b=-1.4737 pixels

Figure 32: Refining Alignment Hypothesis

Iterative Newton-Raphson searches yield good results because I(T) is a well-behaved function, and our best initial alignment hypothesis is often very close to the maximum.

35

The method usually takes 4 to 5 steps to converge. Figure 32 shows a comparison of the initial and refined alignment. There are also other methods for optimization; see [7] for a good account of references.

9. Delaunay Walk

Delaunay triangulation is used heavily as a means for spatial indexing, segmentation, search, and optimization in our SAR ATR system. In this section, we present an algorithm (Delaunay walk) for finding the closest point (peak) with Delaunay triangulation. An immediate application is the association of individual peaks in a test image to peaks in a model image, where the association is done in the sense of closest point. We also provide a theorem as the basis for the algorithm in this section.

9.1 Algorithm

This section describes an algorithm for finding the closest point $q^* \in \mathcal{Q}$ for a given p by using Delaunay triangulation.

To find q^* we start with a randomly selected point, $q \in \mathcal{Q}$, and move to a neighboring point which is the closest to p among q's neighbors, defined by $\mathcal{DT}(\mathcal{Q})$. This locally greedy search process is continued indefinitely until no neighbors of q are closer to p than q. It is proved (Section) that q^* can always be found using this Delaunay walk technique with a random start, and we call this unique property monotonicity of Delaunay triangulation in analogous to the role of monotonic functions in optimization problems. The technique is even more efficient for the problem of finding $q^* \in \mathcal{Q}$ for each $p \in \mathcal{P}$, assuming $\mathcal{P}$ and $\mathcal{Q}$ have the same uniform spatial densities. This increased efficiency is a direct result of using both Delaunay triangulations, $\mathcal{DT}(\mathcal{P})$ and $\mathcal{DT}(\mathcal{Q})$.

Walking on $\mathcal{DT}(\mathcal{Q})$ ...

q_c = a random start point $\in \mathcal{Q}$;
$d_c = \|p - q_c\|$;
continue = yes;
while (continue == yes)
  $\mathcal{NB}$ = neighbors(q_c);
  $q^* = argmin\|p - q\|$, $q \in \mathcal{NB}$;
  $d^* = \|p - q^*\|$;
  if ($d^* < d_c$)
    $q_c = q^*$;
    $d_c = d^*$;
  else
    continue = no;
  end
end
$q^* = q_c$;

Figure 33 shows an example of Delaunay walk. We initialize q_c to q_7 and the algorithm selects the path $q_7 \to q_2 \to q_4 \to q_5$ to reach q_5, which is the closest to p_1.
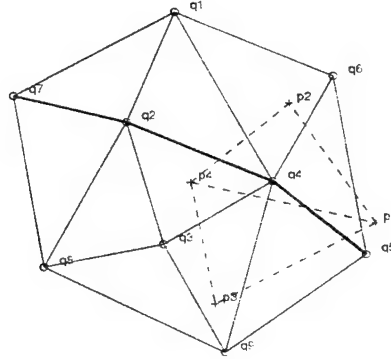

Figure 33: A Delaunay Walk Example

Notice that each q must maintain a list of its neighbors in $\mathcal{DT}(\mathcal{Q})$. The set of all neighbor lists is denoted by $\mathcal{NB}(\mathcal{Q})$. Once $\mathcal{DT}(\mathcal{Q})$ and $\mathcal{NB}(\mathcal{Q})$ are computed, the worst-case search time is proportional to a diameter of $\mathcal{DT}(\mathcal{Q})$. Moreover, except for the first search, the worst-case search can practically be avoided by using $\mathcal{DT}(\mathcal{P})$ and $\mathcal{NB}(\mathcal{P})$. For example, to find the closest $q^* \in \mathcal{Q}$ for p_2, q_c can be initialized to q_5 since p_2 is a neighbor of p_1. In other words, q_c is a point very close to the actual destination q^*, which is q_6 in this case.

9.2 Experiments

The following experiments are done with Matlab and C on a Pentium II 300-MHz machine. Two point sets $\mathcal{P}$ and $\mathcal{Q}$ are generated randomly with a uniform distribution; $\mathcal{P}$ and $\mathcal{Q}$ have the same number of points, i.e., $|\mathcal{P}| = |\mathcal{Q}| = n$. We find the closest $q^* \in \mathcal{Q}$ for each $p \in \mathcal{P}$ with n brute force searches and with n Delaunay walks. The results from both methods are cross-verified. Note that a random start is used only for the first Delaunay walk (Section).

Figure 34 shows the result of the first experiment with Matlab. The CPU time for Delaunay walk includes the time for computing $\mathcal{DT}(\mathcal{P})$, $\mathcal{DT}(\mathcal{Q})$, $\mathcal{NB}(\mathcal{P})$ and $\mathcal{NB}(\mathcal{Q})$. The efficiency of Delaunay walk is evident with larger point sets.

In some applications [3], the data structures $\mathcal{DT}(\mathcal{P})$, $\mathcal{DT}(\mathcal{Q})$, $\mathcal{NB}(\mathcal{P})$ and $\mathcal{NB}(\mathcal{Q})$ are computed once and used many times. Therefore, it is appropriate to compare only the search time of Delaunay walk and brute force search as shown in Figure 35. This experiment is done in C language to optimize the performance. The plot shows that the Delaunay walk outperforms brute force search when n is eight or greater.
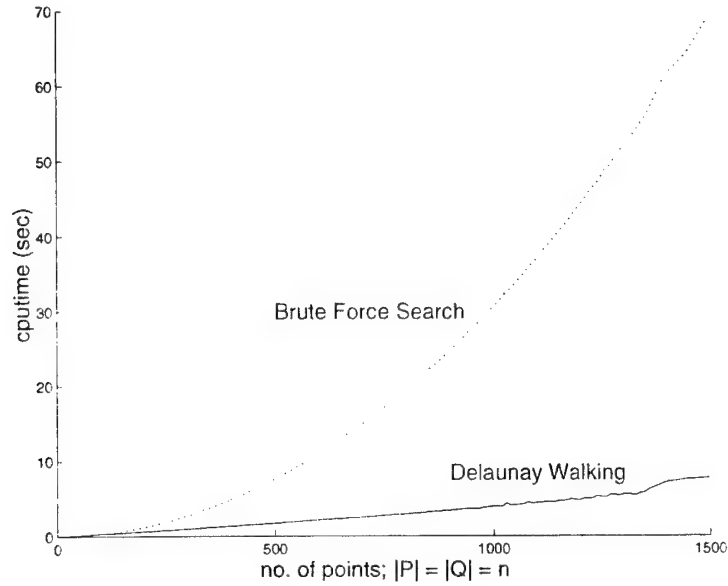
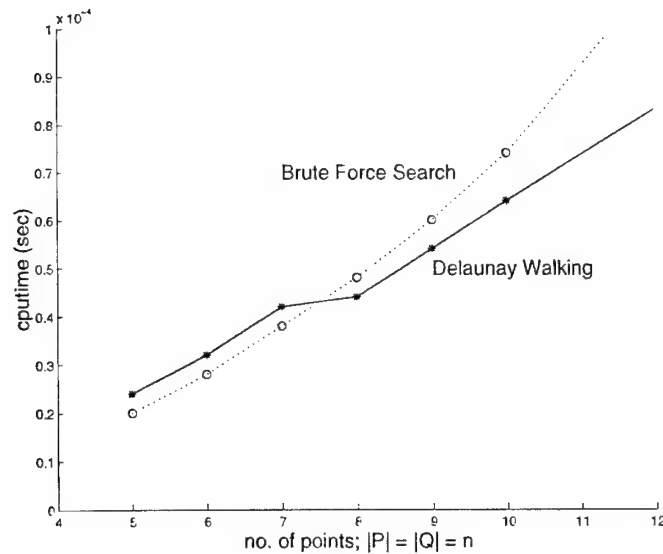Figure 34: Delaunay Walk versus Brute Force Search (Matlab)

Figure 35: Delaunay Walk versus Brute Force Search (C; Search Time Only)

9.3 Theorem and Proof

In this section we present a theorem as the basis for Delaunay walk. This theorem ensures q^* to be found using Delaunay walk with a random start. A proof by contradiction is also provided below.

Theorem:

39

Given a finite point set $Q \subset \mathcal{R}^2$. Let $\mathcal{DT}(Q)$ denote the Delaunay triangulation of Q and $\mathcal{NB}(q)$ the neighbors of $q \in Q$ within $\mathcal{DT}(Q)$. Then for $p \in \mathcal{R}^2$, we have the following implication:

$$\|q^* - p\| \le \|q - p\|, \forall \; q \in \mathcal{NB}(q^*)$$

$$\Rightarrow |q^* - p\| \le \|q - p\|, \forall \; q \in Q \tag{19}$$

In words, if q^* is as close to p as any of q^*'s neighbors, then q^* is a closest neighbor of p over the entire set Q.

Proof:
Let $\mathrm{Cone}(q^*)$ denote the set of all $\triangle's$ in $\mathcal{DT}(Q)$ which have q^* as one of the vertices:
$\mathrm{Cone}(q^*) = \{\triangle abc \in \mathcal{DT}(Q) : \; q^* = a, \; or \; q^* = b, \; or \; q^* = c\}$.

There are two cases for $\mathrm{Cone}(q^*)$:
Case 1. $q^* \in \mathcal{CH}(Q)$;
Case 2. $q^* \notin \mathcal{CH}(Q)$, where $\mathcal{CH}(Q)$ denotes the convex hull of Q.

Now let's proceed with an assumption that q^* is as close to p as any of its neighbors and $\exists \; \hat{q} \notin \mathcal{NB}(q^*)$ such that $\|\hat{q} - p\| < \|q^* - p\|$.
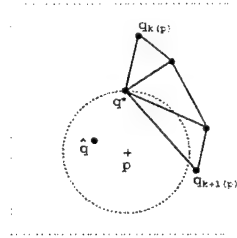

Figure 36: Geometry for the Assumption (Case 1)

Without loss of generality, Figure 36 shows the geometry for the assumption under Case 1. This is, however, not a valid case because the existence of $\hat{q}$ contradicts the fact that $q^* \in \mathcal{CH}(Q)$. To be a valid case, the angle between the two neighboring edges, $\overline{q^* q_{k(p)}}$ and $\overline{q^* q_{k+1(p)}}$, must be smaller than 180 degrees.

Figure 37(a) and (b) fall into the category of Case 2, and they all satisfy the constraint $\angle q_{k(p)} q^* q_{k+1(p)} < 180^o$. But Figure 37(a) is also not possible because $\hat{q}$ is inside $\triangle q^* q_{k(p)} q_{k+1(p)}$, and $\overline{q^* \hat{q}}$ must be present, which contradicts our assumption, $\hat{q} \notin \mathcal{NB}(q^*)$.

The four points, q^*, $q_{k(p)}$, $\hat{q}$, and $q_{k+1(p)}$, form a convex quadrilateral in Figure 37(b). There are two ways to triangulate these four points, as demonstrated in Figure 38(a) and (b). We show below that the triangulations shown in Figure 38(a) do not make a legal Delaunay triangulation, and the proof is completed since 38(b) contradicts the assumption, $\hat{q} \notin \mathcal{NB}(q^*)$.
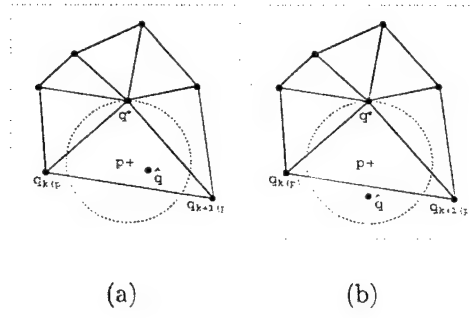
40

Figure 37: Geometry for the Assumption (Case 2)
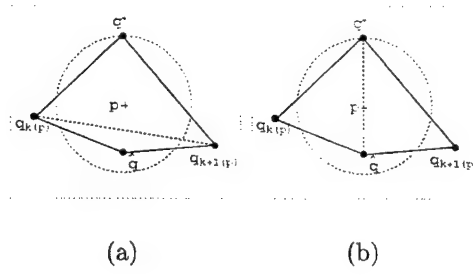
Figure 38: Two Possible Triangulations of a Convex Quadrilateral
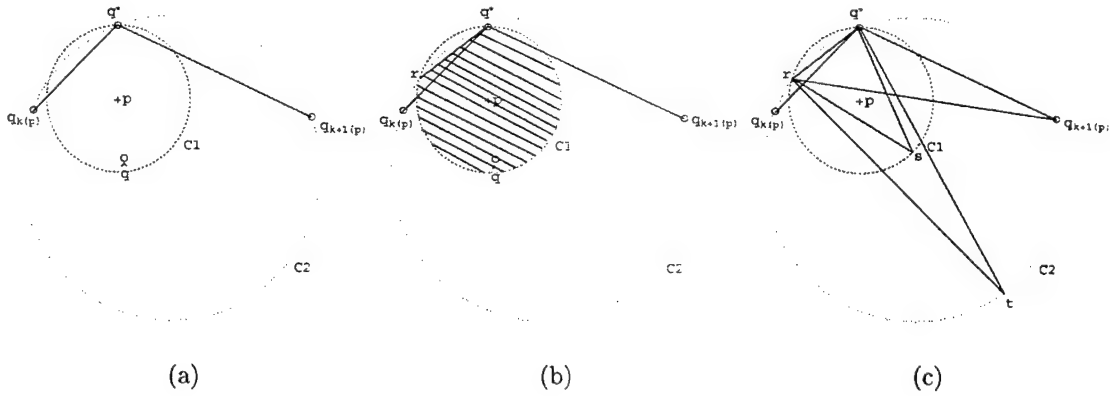
Figure 39: Geometry of the Proof

Figure 39(a) shows the geometry of the problem. C_1 is the circle determined by q^* and p, and C_2 is the circle determined by q^*, $q_{k(p)}$, and $q_{k+1(p)}$. Note that points $q_{k(p)}$, and $q_{k+1(p)}$ are outside C1 by the assumption. If C_1 and C_2 intersect only at q^*, then C_1 is completely inside C_2. If C_1 and C_2 intersect at two points, q^* and r, then the hatched region in Figure 39(b) is inside C_2.

The above statement may be trivial. But in order to be more rigorous, we can use Thales'

theorem [8] to prove this is indeed the case. Because C_1 and C_2 share the same cord $\overline{rq^*}$, and $q_{k+1(p)}$ is outside C_1, by Thales' theorem we have
$\angle rsq^* > \angle rq_{k+1(p)}q^* = \angle rtq^*$, $\forall(s \in C_1, t \in C_2)$ on the lower side (the side containing p) of $\overline{rq^*}$. Therefore, the hatched region, is inside C_2. Since our $\hat{q}$ is always in the hatched region, which is completely inside C_2, $\overline{q_{k(p)}q_{k+1(p)}}$ is an illegal edge because the circle C_2 is not site-free [8]. Furthermore, because q^*, $q_{k(p)}$, $\hat{q}$, and $q_{k+1(p)}$ form a convex quadrilateral, exactly one of $\overline{q_{k(p)}q_{k+1(p)}}$ and $\overline{q^*\hat{q}}$ is legal in general situations, i.e., when the four points are not on a common circle [8]. In conclusion, Figure 38(a) is not possible and $\overline{q^*\hat{q}}$ must be present.

10. Target Matching

We use synthesized intensity images (synthesized from peaks) and a probability matching metric for both image alignment and target recognition. Our preliminary experiments achieved good recognition rates for test images with different depression angles. Recognition of targets with different depression angle and configuration from the training set is also tested for T72.

10.1 Matching Metrics

Let U, $D_U = \{(i,j)|U_{ij} > 0\}$ and V, $D_V = \{(i,j)|V_{ij} > 0\}$ be the synthesized intensity image of the template and test image respectively. Where $U_{ij} = \sum_k A_{uk} e^{-\frac{(x_{uij}-x_k)^2+(y_{uij}-y_k)^2}{2\sigma^2}}$, $V_{ij} = \sum_k A_{vk} e^{-\frac{(x_{vij}-x_k)^2+(y_{vij}-y_k)^2}{2\sigma^2}}$ and $\sigma \sim 1$ pixel.

A natural deterministic matching metric is the Euclidean distance of the two image vectors in multidimensions:

$$\sqrt{\sum_{(i,j)\in D}(V_{ij} - U_{ij})^2}, \ D = D_V \cup D_U \tag{20}$$

Each pixel intensity difference, $V_{ij} - U_{ij}$, is modeled simply as a zero mean Gaussian random variable with a standard deviation depending on the mean pixel intensity; i.e., $X_{ij} = V_{ij} - U_{ij} \sim N(0, \sigma_{ij})$, where $\sigma_{ij} = \sigma_{ij}(\frac{V_{ij}+U_{ij}}{2})$. We use a linear model for σ_{ij}, $\sigma_{ij} = \eta \cdot \frac{V_{ij}+U_{ij}}{2} + \epsilon$. A normalized measure is $Y_{ij} = \frac{X_{ij}}{\sigma_{ij}} \sim N(0,1)$. Therefore, a normalized distance measure has the following form:

$$\sqrt{\sum_{(i,j)\in D}Y_{ij}^2}, \ D = D_V \cup D_U \tag{21}$$

Note that $\sqrt{\sum_{(i,j)\in D}Y_{ij}^2}$ or $\sum_{(i,j)\in D}Y_{ij}^2$ is a probability measure because $Pr(distance > \sqrt{\sum_{(i,j)\in D}Y_{ij}^2})$ can be used as a matching metric: the larger the probability, the more likely the match. Ideally, if the template image and the test image come from the same target, then $D \approx D_U$ and under *i.i.d.* assumption $\sum_{(i,j)\in D}Y_{ij}^2$ will have a Chi-square distribution with $|D_U|$ (size of D_U) degrees of freedom. In order to take into account the effect of $|D_U|$, the following matching metric is used:

$$\frac{1}{2\sqrt{|D_U|}} \sum_{(i,j) \in D} (Z_{ij} - 1) \longrightarrow Z \sim N(0,1), \; where \; Z_{ij} = Y_{ij}^2 \tag{22}$$

Central Limit Theorem is used in Equation (22); it is applicable since $|D_U| \sim 500$. Note also that $mean(Z_{ij}) = 1 \; and \; var(Z_{ij}) = 2$.

10.2 Experiments and Results

The target database contains a total of 300 template images; 100 for each of the 3 targets: BTR70, BMP2, and T72. The template images have a 17-degree depression angle. They are chosen to span evenly the 0 to 360 degrees target orientation range. For each test image, we perform peak detection, segmentation, and leading surface estimation, in that order. Only the test images with correctly estimated orientation (i.e., error 10 degrees) are used in the recognition experiments.

Estimated target orientation, α, of the test image is used to select N images (from each target class) that have the closest orientation to α as hypotheses. N images (from each target class) that have the closest orientation to $\alpha \pm 180^o$ are also selected as 180-degree alternative hypotheses. Alternative hypotheses are needed because leading surfaces only determine target orientation up to a 180-degree flip. Expression (22) is a measure of matching disparity. The hypothesis that has the lowest matching disparity determines the class name and orientation of the test target. Note that N=3 is used in our experiments.

Expression (22) in is not only used as a target matching metric, but also as an image alignment metric. In other words, expression (22) is optimized for a given image pair during the process of image alignment. The metric is a function of the transformation T; therefore, the aligning method in Section is applicable here. However, aligning with expression (22) is computationally more expensive compared to aligning with ideogram.

The confusion matrix for test images with depression angle = 17 degrees is shown in Table 9. For the BTR70, there is a total of 118 test images; 116 of them are correctly recognized as BTR70, and 2 of them are recognized as 180-degree alternative BTR70 (i.e., wrong pose). Figure 40 shows the matching disparities of the hypotheses as a function of target orientation. The disparity of the BTR70 hypothesis is used as a reference. The overall recognition rate for the three targets is 98.9 percent. Table 10 shows the confusion matrix for test images with depression angle = 15 degrees. The overall recognition rate for the three targets is 95.8 percent.

44

Table 9: Test Depression Angle = 17°

	BTR70	BMP2	T72
BTR70 (98.3%)	116/2	0/0	0/0
BMP2 (98.4%)	1/1	121/0	0/0
T72 (100%)	0/0	0/0	116/0

Table 10: Test Depression Angle = 15°

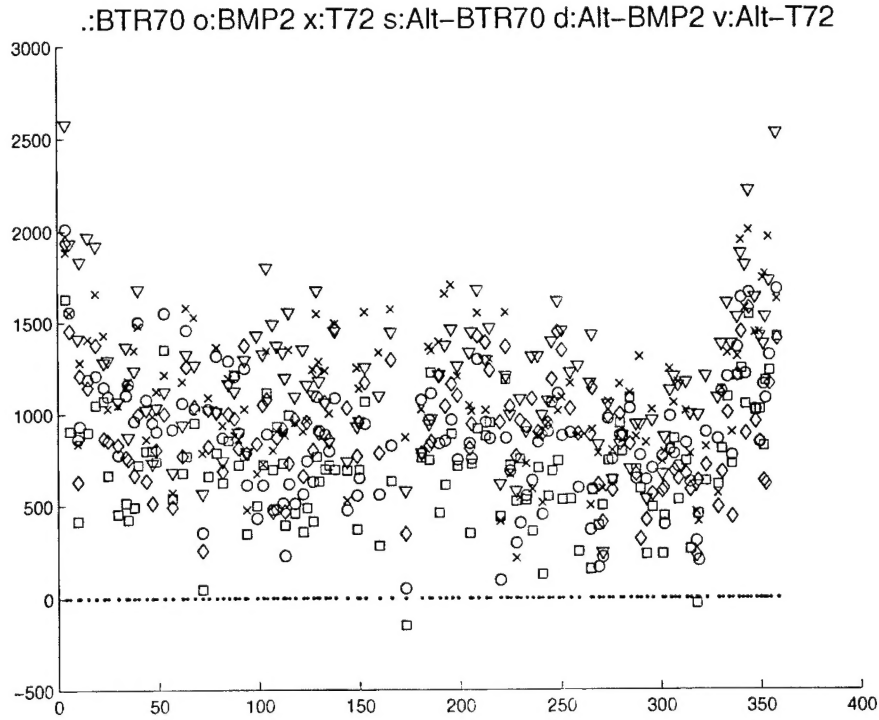|	BTR70	BMP2	T72
BTR70 (95.2%)	178/9	0/0	0/0
BMP2 (96.2%)	1/0	178/3	3/0
T72 (96.1%)	0/0	5/0	171/2

Figure 40: Matching Disparities

References

[1] B. Wang, "An Automatic Target Recognition System for SAR Imagery," PhD Thesis, Stanford University, 1997.

[2] L. M. Novak et al., "Radar Target Identification Using an Eigen-image Approach," *Proceedings*, IEEE National Radar Conference pp. 129–131, 1994.

[3] Tsung-Liang Chen and Thomas O. Binford, "Context and Quasi-Invariants in ATR with SAR Imagery," *Proceedings*, Image Understanding Workshop, vol. #2, pp. 839-848, 1998.

[4] D. Dudgeon et al., "Use of Persistent Scatterers for Model-based Recognition," *SPIE*, vol. #2230, pp. 356–368, 1994.

[5] T. O. Binford et al.,"Context and Quasi-Invariants in ATR with SAR Imagery," *Proceedings*, Image Understanding Workshop, 1997.

[6] Oliver, C.J., Blacknell, D., and White, R.G., "Optimum Edge Detection in SAR," *IEE Proceedings - Radar, Sonar and Navigation*, vol. #143, no.1, pp. 31-40, 1350-2395, IEE Feb. 1996.

[7] W. H. Press et al., *Numerical Recipes in C*, 2nd Edition, Cambridge University Press, 1992.

[8] Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf, *Computational Geometry: Algorithms and Applications*, Springer-Verlag, 1997.

[9] K. Ikeuchi et al., "Model-based SAR ATR System," *Proceedings*, Image Understanding Workshop, vol. #2, pp. 1263–1276, 1996.

[10] E. R. Keydel et al., "MSTAR Extended Operating Conditions: A Tutorial," *SPIE*, vol. #2757, pp. 228–242, 1996.

[11] L. M. Novak et al.,"ATR Performance Using Enhanced Resolution SAR," *SPIE*, vol. #2757, pp. 332–337, 1996.

[12] A. Mahalanobis et al., "A Quadratic Distance Classifier for Multi-Class SAR ATR using Correlation Filters," *SPIE*, vol. #1875, pp. 84–94, 1993.

[13] S. M. Verbout et al., "Improving a Template-Based Classifier in a SAR Automatic Target Recognition System by Using 3-D Target Information," *Lincoln Laboratory Journal*, vol. #6 no. 1, pp. 53–76, 1993.

[14] S. D. Halversen, "Calculating The Orientation of A Rectangular Target in SAR Imagery," *IEEE* 1992 National Aerospace and Electronics Conference, vol. #1, pp. 260–264, 1992.

[15] H. J. Wolfson et al., "Model-based Object Recognition by Geometric Hashing," First European Conference on Computer Vision Proceedings, pp. 526–536, 1990.

List of Acronyms

ACRONYM	DESCRIPTION
ATR	Automatic target recognition
edgels	Edge elements
EOCs	Extended operating conditions
ISAR	Inverse synthetic aperture radar
LLS	Longer leading surface
SLS	Shorter leading surface
MSTAR	Moving and stationary target acquisition and recognition
ROI	Region of interest
SAR	Synthetic aperture radar